BALANCING AND RATIO EDITING WITH THE NEW SPEER SYSTEM

Lisa R. Draper and William E. Winkler
{lisa.r.draper,william.e.winkler}@census.gov

ABSTRACT

This paper describes theory, computational algorithms, and software associated with the new SPEER edit system. The SPEER edit system is based on the Fellegi-Holt model (JASA, 1976) of editing and is used on continuous data. The key feature of the new SPEER system is that it automatically does ratio editing and a limited form of balancing (assuring the items add to totals). The limited form of balancing appears to work in over 99% of the situations in which balancing is needed and the associated computational algorithms are exceedingly fast. Other economic edit systems are not able to do automatic balancing in a manner that assures records satisfy all edits.

Keywords: economic data, error localization

1. INTRODUCTION

Economic data in administrative or survey files may contain large numbers of records, some of which contain logical inconsistencies or incorrect data. Errors can arise because methods of creating records in files are not consistent, because questions are not understood, or because of transcription or coding problems. In many situations, data files are edited using custom software that incorporates rules developed by subject-matter specialists. If the specialists were unable to develop the full logic needed for the edit rules, then the subsequent edit software would be in error. If programmers do not properly code the rules, then the software would be in error. Developing software from scratch each time a database is redesigned is time-consuming and error-prone. It is better to have a system that can describe edit rules in tables that are read and utilized by reusable software modules. The tables could be more easily updated and maintained than complex if-then-else rules in computer code. The software would automatically check the logical validity of the entire system prior to the receipt of data during production processing.

Fellegi and Holt (1976), hereafter referred to as FH, provided the theoretical basis of such a system. FH had three goals that we paraphrase:

1. The data in each record should be made to satisfy all edits by changing the fewest possible variables (fields).
2. Imputation rules should derive automatically from edit rules.
3. When imputation is necessary, it should maintain the joint distribution of variables.

The key to the FH approach is to understand the underpinnings of goal one. Goal one is referred to as the *error localization* problem. In the FH model, a subset of the edits that can be logically derived from the explicitly defined edits (called *implied or implicit edits*) are needed if the error localization problem is to be solved. FH provided an inductive, existence-type proof to their Theorem 1 that demonstrated that it is possible to find the region in which the error localization problem could be solved. Their solution, however, did not deal with many of the practical computational aspects of the problem.

SPEER, or Structured Programs for Economic Editing and Referrals, was originally developed by Brian Greenberg (e.g., Greenberg and Surdi, 1984). It consisted of two modules: one for generating the implicit edits and the other for error localization and imputation. *Error localization* is the process of determining the minimum number of fields that must be changed in an edit-failing record so that the record satisfies all edits. The new SPEER system consists of four modules, two main modules similar to those in the earlier SPEER and two auxiliary modules. One key new feature is the auxiliary module in SAS (Statistical Analysis System) for automatically determining bounds for the ratio edits (Thompson and Sigman, 1996). The second new feature is a simple form of balancing that is implemented in the error localization module. The balancing algorithm holds for the overwhelming majority of balance situations that are encountered with actual survey data. Further details of the new SPEER system are given later in this paper.

This paper's main result is an algorithm for single-level balancing that works simultaneously with ratio edits. By *single-level balancing*, we mean that an item (field) can appear in at most one balance equation. Based on a review of more than 100 Bureau of the Census economic surveys, 99% of items appear in no balance equations or in single-level balance equations only.

The outline of this paper is as follows: In the second section, we give notation, background material, and an overview of the new SPEER system. The third section presents our algorithm that combines single-level balancing with ratio editing. The algorithm is used in the error-localization module and is very efficient computationally. In the fourth section, we provide some empirical results from a computer system (Winkler and Draper, 1997) that is based on the new theory and algorithms. The final two sections consist of discussion and summary.

2. **BACKGROUND AND NOTATION**

The goals of the new SPEER system are (1) theoretical validity, (2) exceptional speed, (3) nearly automatic determination of error bounds, (4) passing edits and satisfying balance equations after one pass through the data, and (5) straightforward maintenance by good FORTRAN programmers. Since the original version written by Brian Greenberg, SPEER has been theoretically valid and exceptionally fast. The new version has entirely new algorithms and source code that are designed to be easily maintained by good programmers. New SPEER is only half as fast as the original SPEER. The speed decrease is due to the balancing algorithms. The current version of SPEER has bound determination due to Thompson and Sigman (1996) that uses the Exploratory Data Analysis (EDA) method of resistant fences. Analysts who used SPEER indicated they wanted a method for determining the bounds used in the ratio edits and they needed specified items to add to totals. Presently, SPEER is the only editing system for continuous data to assure balancing and to give a means of determining bounds.

If variables are defined by V_i , i = 1, ..., N, then ratio edits take the form:

$$L_{ij} < V_i / V_j < U_{ij} \tag{2.1}$$

and balance edits take the form

$$\sum_{i \in S} V_i - V_j = 0, \tag{2.2}$$

where S is a proper subset of the first N integers and $j \notin S$. Simple algebra allows the re-expression of the two ratio inequalities in (2.1) as two linear inequality edits and the equality in (2.2) as two linear inequality edits. The bounds L_{ij} and U_{ij} can be determined by analysts through use of prior data.

FH (Theorem 1) established that, if we start with a subset of the fields that satisfy all edits that place restrictions on those fields only, then it is possible to fill-in the remainder of the record with values in the remaining fields so that the record satisfies all edits. To be more precise, if a record has n fields and we assume that we are starting with k fields, then we can find a value for field k+1 so that the record satisfies all edits on the first k+1 fields. If we are in the process of imputing a value for field k+j+1, then we say that the first k+j fields have been *established*. The ordering in which we fill-in fields (i.e., impute) affects the values that can be imputed for fields k+j+1. In the earliest versions of SPEER which only used ratio edits, the edit bounds and the values in the first k+j fields created restraints that yielded an interval (or point) into which the value of the $k+j+1^{st}$ field had to be imputed if edits were to be satisfied. In the current version of SPEER, the balance equations place further restraints on the intervals into which the $k+j+1^{st}$ field can be imputed.

We note that the bound L_{ij} is the largest lower bound on V_i / V_j and U_{ij} is the smallest upper bound on V_i / V_j for equation (2.1). For simplicity of our illustration, we assume that the equation

$$V_1 + V_2 = V_3$$

needs to hold. In all situations, we will only create new implicit edits by combining ratio edits with other implicit edits that are needed. We refer to the left hand side (LHS) of the balance equation as the side that contains items to be added and the right hand side (RHS) as the total. Similarly, we refer to the LHS of an implicit edit induced by a balance equation and one or more ratio edits as the side that contains two or more terms and the RHS as the side of the inequality that only contains one term. Implicit linear inequality edits that are obtained by replacing terms on the LHS of a balance equation with the appropriate terms from a ratio inequality are the main set of implicit edits with which we will be concerned. We will show that a subset of the implicit edits is typically needed for error localization (EL). The subset yields solutions for virtually all of the situations that we encounter with actual survey data. The subset is far more easily computed than the full set of implicit edits. We call the subset of implicit edits *induced* edits. Further, we will show that only a subset of the induced edits, those induced by a balance equation and a single replacement of a term on the LHS, are needed for computing the intervals into which items can be imputed. The latter result is particularly important because the code associated with the algorithm for determining the interval into which to impute is not particularly easy. If the terms in the balance equation do not add to the total, we say that the balance equation fails. If the ratio of two variables is greater than the upper bound or is less than the lower bound, we say that the ratio edit has failed. We say that an edit is *satisfied* if the edit does not fail.

SPEER allows individual fields to be restrained by at most one balance equation, which we refer to as *single-level balancing*. Extensive review of the edits in use for economic surveys at the Census Bureau has shown that well over 99% of fields in different surveys need to be restrained by at most one balance equation. Whereas creating algorithms and writing software for general, multi-level balancing has never been accomplished, the algorithms and computer code associated with the one level of balancing in SPEER are reasonably straightforward.

SPEER FORTRAN software consists of three main programs. The first generates implicit edits (bounds) and checks the logical consistency of the ratio edits only. An auxiliary simplex program (in

SAS) checks the logical consistency of the set of ratio and balance edits. The second program generates regression coefficients for the equation $V_1 = ß_{12} V_2 + \varepsilon$ that are used in the imputation module of the main SPEER program. The main SPEER program also uses the implicit edits and the raw data file as inputs. Prior to imputation, the main SPEER program generates failed induced edits that can be derived from combinations of ratio and balance edits.

Due to the simplicity of algorithms, SPEER code is exceedingly fast. Generating 272 pairs of implicit edit bounds in each of 546 industrial categories requires a total of 35 seconds on a SPARC station 20 and 115 seconds on a 75 MHZ Pentium. With Annual Survey of Manufactures data having 17 fields, 136 ratio edits, and 2 single-level balance equations, SPEER needed 70 seconds (wall clock time) to edit 5000 records on a 200 MHZ Pentium Pro and 9 minutes (wall clock time) to edit 9765 records on a VAX 6000 system running under VMS. Because ratio edits are inherently straightforward, most SPEER code is easy to understand and maintain. The code is completely portable. Using SPEER on other machines merely requires copying FORTRAN source code and recompiling it.

3. **THEORETICAL RESULTS**

This section consists of several lemmas, a theorem, and the main algorithm. To better understand the main algorithm, we provide an additional description of the edit/imputation module. The earlier version of the SPEER edit/imputation only used ratio edits. The minimal number of fields to impute and the intervals into which to impute were straightforward to compute. The new SPEER first checks if a ratio edit or balance equation fails. If there is a failure, then the appropriate induced edits are computed and checked in the main edit/imputation module (i.e., "on the fly"). Implicit edits based on combining ratio edits and balance equations are not computed a priori. The failing induced edits, failing ratio edits, and failing balance equations determine the fields and equations that are used in the error localization (EL) algorithm that determines the minimal number of fields to impute. We use a greedy algorithm (Nemhauser and Wolsey, 1987) to determine the minimum number of fields to impute. With one exception, the imputation intervals into which values can be imputed are determined by the ratio edits and induced edits only. The only time that the balance equations are used is the one exception, when all but one item in a balance equation is known.

In the following, we assume that all fields can be connected (paired) with other fields via ratio edits and that all fields in a balance equation are restrained by ratio edits. Our assumption means that we deal with the only difficult situation involving combinations of ratio edits and balance equations. If one or more items in a balance equation were not restrained by ratio edits, then we could drop the balance equation from consideration in the main SPEER module because balancing could be easily dealt with after running SPEER. The ratio restraints in SPEER could be used to impute the items in the balance equations and the balance equation, if necessary, could be used to impute one of the items not restrained by the ratio edits.

In the following, we will typically replace a term in a balance equation of the form

$$V_1 + V_2 = V_3 \tag{3.1}$$

to get an implicit edit of the form

$$U_{1j} V_j + V_2 \geq V_3 \tag{3.2}$$

from the appropriate ratio inequality

$$U_{1j} V_j \geq V_1 . \tag{3.3}$$

Implicit edits that are derived by replacing terms in a balance equation with appropriate terms from ratio edits will be called *induced* edits. If an induced edit is obtained by replacing only one term in a balance equation with the appropriate terms from a ratio edit, it will be called a *simple induced* edit; otherwise, a *nonsimple induced* edit. Simple induced edits give the most information needed for determining intervals into which values of variables can be imputed. For instance, if the EL solution includes V_1 and V_2, then the simple induced edit (3.2) gives us important information. If we change values of V_1 and V_2 appropriately to assure that (3.2) is satisfied, then both the balance equation (3.1) and the ratio edit (3.3) will necessarily be satisfied. In other words, the simple induced edits give us the best information for determining the intervals into which we need to impute. As shown by FH, we need virtually all of the implicit edits to determine the EL solution. The goal of this section will be to show that an appropriately chosen subset of the induced edits will allow us to determine virtually all EL solutions that are needed with actual survey data. There is a small proportion of records that our methods do not allow us to error localize directly. They can be dealt with via a heuristic that we propose. The crucial advantage of these methods is that they are much faster, are much easier to apply in most survey situations, and yield more easily maintained code than methods that rely on more general linear inequality edits such as Statistics Canada's Generalized Edit and Imputation System (GEIS). Kovar and Winkler (1996) did a direct comparison of GEIS and an earlier version of SPEER that had primitive balancing algorithms.

The following lemma tells us that if we have replaced a term on the LHS on the balance equation with the appropriate bound (either U_{ij} or L_{ij}) and variables, then we do not need to do a second replacement on that term.

Lemma 3.1. Assume that $j \neq 1$, $j \neq k$, and $k \neq j$. Then, implicit edit $U_{1k} U_{kj} V_j + V_2 \geq V_3$ is redundant to induced edit $U_{1j} V_j + V_2 \geq V_3$ and implicit edit $L_{1k} L_{kj} V_j + V_2 \leq V_3$ is redundant to induced edit $L_{1j} V_j + V_2 \leq V_3$.
Proof. By construction the ratio edit $U_{1k} U_{kj} V_j \geq V_1$ is redundant to $U_{1j} V_j \geq V_1$. In other words, whenever the edit determined by the second inequality in the previous sentence is satisfied, the first inequality is also satisfied. If the induced edit $U_{1j} V_j + V_2 \geq V_3$ is satisfied (the inequality does not fail), then implicit edit $U_{1k} U_{kj} V_j + V_2 \geq V_3$ is always satisfied.

It is straightforward to extend Lemma 3.1 and other results of this section to balance equations and inequalities with more than three terms. The following lemma shows that we do not need to consider implicit edits that are induced by replacing the RHS of a balance equation or induced edits of the type considered in this section with the appropriate terms from a ratio edit.

Lemma 3.2. Implicit edits of the forms $V_1 + V_2 \geq L_{3k} V_k$ and $V_1 + V_2 \leq U_{3k} V_k$ are not needed for determining the interval into which to impute.
Proof. As an upper bound on V_k, the inequality $V_1 + V_2 \geq L_{3k} V_k$ is always redundant to the ratio edit $V_3 \geq L_{3k} V_k$. As a lower bound on V_1, $V_1 \geq L_{3k} V_k - V_2$ is always redundant to $V_1 \geq L_{1k} V_k$. In other words, whenever we assure that the bounds determined by the appropriate ratio edits are satisfied, then the bounds determined by the implicit edits of this lemma are automatically satisfied.

Other variables and directions are done similarly.

We observe that the method of proof also yields that the implicit edits that are obtained from replacing the RHS of an induced edit, $U_{1j} V_j + V_2 \geq L_{3k} V_k$ and $L_{1j} V_j + V_2 \leq U_{3k} V_k$, are not needed for determining the interval into which to impute. The following lemma yields an important reduction in computation and simplification of algorithms because it tells us that we only need to consider failing ratio edits when we look for failing induced edits. If an induced edit fails, then it was necessarily generated by a failing ratio edit or generated by either a failing induced edit or failing balance equation.

Lemma 3.3. A failing induced edit that is implied by a failed balance equation and a non-failing ratio edit is not needed for determining the interval into which to impute.
Proof. Assume that the induced edit $U_{1j} V_j + V_2 \geq V_3$ fails and the appropriate $U_{1j} V_j \geq V_1$ ratio edit does not fail. Then $V_1 + V_2 = V_3$ fails in such a manner that $V_1 + V_2 < V_3$ and $U_{1j} V_j + V_2 < V_3$. If V_3 is part of the error-localization solution and V_1 and V_2 have already been established, then V_3 must be decreased in amount until both $U_{1j} V_j + V_2 \geq V_3$ and $V_1 + V_2 = V_3$ are satisfied. Since the difference (V_3 -V_1 - V_2) is greater than the difference (V_3 - $U_{1j} V_j$ - V_2), the upper bound on the change in V_3 determined by the desired balance equation always has precedence over the bound in the change in V_3 determined by the induced edit. In other words, if we assure that the balance equation is satisfied, then the induced edit is automatically satisfied. If V_2 is part of the error-localization solution and V_1 and V_3 have already been established, then V_2 can be determined in a similar manner. The induced edit is also redundant. The remaining cases are dealt with similarly.

Lemma 3.3 is important because if we extend its reasoning, it tells us that a failing induced edit that is associated with a ratio edit is likely to be more important than a failing induced edit that is associated with a non-failing ratio edit. This yields a large reduction in computation because we only consider induced edits that are with small subset of failing ratio edits rather than the entire set of all ratio edits.

The main theorem of FH shows that it is always possible to find a set of fields S that can be changed so that no edits (explicit and implicit) fail. FH actually showed that, if set S contains at least one variable (entering field) from each failing edit, then S will provide an EL solution. Typically, when we refer to the error localization solution, we mean the minimum number of fields that must be changed so that all edits no longer fail. Necessarily, we must change at least one variable (field) in every failing edit so that the edit no longer fails. By the reasoning similar to that used in proving Lemma 3.3, a failing non-simple induced edit is one that is derived from a failing induced edit and a failing ratio edit. We need not consider non-failing ratio edits. Lemma 3.4 shows that all induced edits are needed for error localization.

Lemma 3.4. The induced edits of the forms $U_{1j} V_j + U_{2k} V_k \geq V_3$ and $L_{1s} V_s + L_{2t} V_t \leq V_3$ are needed for error localization.
Proof. Assume that the two ratio edits $U_{1j} V_j \geq V_1$ and $U_{2k} V_k \geq V_2$ and the two induced edits $U_{1j} V_j + V_2 \geq V_3$ and $V_1 + U_{2k} V_k \geq V_3$ have all failed. The balance equation $V_1 + V_2 = V_3$ may or may not fail. If the induced edit $U_{1j} V_j + U_{2k} V_k \geq V_3$ is not needed for error localization, then V_1 and V_2 is a potential solution because V_1 and V_2 are in every failing edit. However, if the induced edit $U_{1j} V_j + U_{2k} V_k < V_3$ (i.e., has failed), then we must change at least one of V_j , V_k , and V_3. This

contradiction shows that $U_{1j}\ V_j + U_{2k}\ V_k \geq V_3$ is needed for error localization. The need of the induced edit $L_{1s}\ V_s + L_{2t}\ V_t \leq V_3$ is proved similarly.

Another way of thinking about the need for the edit $U_{1j}\ V_j + U_{2k}\ V_k \geq V_3$ is the following. Assume that V_3 is part of the error localization solution and that $U_{1j}\ V_j + U_{2k}\ V_k < V_3$ and the associated ratio edits have both failed. We must change V_3 until it is smaller than $U_{1j}\ V_j + U_{2k}\ V_k$. Assuring that V_3 is smaller than both $U_{1j}\ V_j + V_2$ and $V_1 + U_{2k}\ V_k$ is not sufficient.

The following theorem yields significant simplifications in the algorithms for computing the intervals into which values can be imputed.

Theorem 3.1. The simple induced edits are sufficient for determining the intervals into which items can be imputed.
Proof. Assume that the non-simple induced edit $U_{1j}\ V_j + U_{k2}\ V_k \geq V_3$ fails. Then we have five inequalities representing edit failures: $U_{1j}\ V_j + U_{2k}\ V_k < V_3$, $V_1 + U_{2k}\ V_k < V_3$, $U_{1j}\ V_j + V_2 < V_3$, $U_{1j}\ V_k < V_3$, and $U_{2k}\ V_k < V_2$. We only consider the case when at least two of V_1, V_2, and V_3 must be imputed. Assume V_2, and V_3. Also, assume that V_j and V_k are established (i.e., we are using their original values because they are not part of the error-localization solution or they have already been imputed). If we first impute V_2, then the induced edit $U_{1j}\ V_j + U_{k2}\ V_k \geq V_3$ is not needed. Among the five inequalities, only the ratio edit $U_{2k}\ V_k < V_{2.}$ is needed for determining the interval into which we can impute. If we now first impute V_3, then we must adjust V_3 downward so that the inequalities (failed induced edits) $U_{1j}\ V_j + U_{2k}\ V_k < V_3$ and $V_1 + U_{2k}\ V_k < V_3$ are not valid. Since V_1 is not being imputed, the ratio inequality $V_1 \leq U_{1j}\ V_j$ holds. Because $V_1 + U_{2k}\ V_k \leq U_{1j}\ V_j + U_{2k}\ V_k$, only the inequality $V_1 + U_{2k}\ V_k < V_3$ from the failure of the simple induced edit is needed to determine the imputation interval for V_3. Other situations are dealt with similarly.

The algorithm used in the new SPEER is:

1. If any ratio or balance equations are failed, then compute induced edits and determine which of them have failed.
2. Use the failed ratio, balance, and induced edits in a greedy algorithm to determine the number of fields to impute.
3. For each field that must be imputed, first determine whether the value of the field can be determined by a balance equation. If it can be, do so. If it cannot, then use ratio edits and simple induced edits to determine an interval into which the imputed value for the field must be imputed via the chosen imputation method.
4. Determine whether the ratio imputation lies in the proper interval. If it does use it; otherwise, choose a value slightly above the lower bound of the interval if the original value of the field is less than the lower bound or choose a value slightly below the upper bound if the original value in the field is above the upper bound.

In the SPEER system, we use a greedy algorithm rather than more general methods such as branch-and-bound. A greedy algorithm will not find minimal number of fields to impute. The main reason that the greedy algorithm is used is that it is often hundreds or thousands of times faster than branch and bound which is the known best general way of finding optimal solutions (Nemhauser and Wolsey, 1987).

4. EMPIRICAL RESULTS

The current version of SPEER has straightforward algorithms that allow it to determine which ratio edits, balance equations, and simple induced edits have failed. These failing edits are used to determine the fields (items) in the error localization solution and the intervals into which the items can be imputed. As we know from the theoretical development in Section 3 (in particular, Lemma 3.4), simple induced edits are not sufficient for error localizing all possible combinations of errors. As we believe that most errors in the survey data are not too serious, we examine how many errors can be automatically made to satisfy edits with the existing algorithms if we do multiple passes against the data. After the first pass through SPEER, a small proportion of records will only be partially corrected and fail a smaller number of edits than they failed originally. If we pass these semi-corrected records through SPEER a second a time, then they are more likely to pass all edits. Our procedure is to pass records through SPEER multiple times, determine how many records fail after each pass, and examine the types of errors that remain after each pass. The preliminary set of passes will tell us if a moderate expansion of the algorithms in SPEER is likely to yield a system in which a high proportion (99+%) pass all edits after one or two passes. We note that a moderate expansion of the algorithms will still yield a SPEER that is exceedingly fast (e.g., Winkler and Draper, 1997, Kovar and Winkler, 1996).

The data used in the empirical study is keyed data from the 1995 Annual Survey of Manufactures. The responses are collected on a 4-page paper questionnaire. The fields edited in the SPEER program consist of 17 fields, defined as follows:

SW	Salary and Wages (WW+OW)
VS	Value of Shipments
TE	Total Employment (PW+OE)
WW	Production Worker Wages
OW	Other Employee Wages
TIB	Total Inventory - Beginning of Year
CM	Cost of Materials
TIE	Total Inventory - End of Year
PW	Number of Production Workers
OE	Number of Other Employees
PH	Number of Plant Hours Worked
LE	Legally Required Fringe Benefits
VP	Voluntarily Paid Fringe Benefits
PTIE	Calculated Sum of Details of TIE
PTIB	Calculated Sum of Details of TIB
PVS	Calculated Sum of Details of VS
PCM	Calculated Sum of Details of CM

Additive fields include Salary and Wages (which is the sum of Production Worker Wages plus Other Employee Wages), and Total Employment (which is the sum of Number of Production Workers plus Number of Other Employees).

The last four fields are referred to as pseudo totals. They contain the calculated sum of the detail items for each of their corresponding totals. Pseudo total testing is useful because it enhances the ratio edit's ability to choose a reported total or reported sum of details when the two items differ. The explicit ratio edits are defined by the subject matter experts. These ratios are run through a bounds-generating program that produces the appropriate set of ratio bounds for every possible combination of fields. These are known as the implicit ratio edits and are easily computed.

The results from applying the version of SPEER that can only deal with first-level induced edits are given in Table 1. Virtually all of the 175 records that fail edits after the second pass are from records that fail non-simple induced edits of the form given in Lemma 3.4 on the first pass. Results from applying the version of SPEER that is able to deal with second-induced edits and has a heuristic are given in Table 2. Most of the 43 records first-pass failing edits fail two of the second-level induced edits and have their balance equations (and second-level induced edits) connected by a ratio edit.

Table 1. Results from Different Passes
Through the SPEER System
First-Level Induced Edits Only
9,765 Records

Pass	Failed	Passed
First	5,343	4,422
Second	721	9,044
Third	175	9,590

Table 2. Results from Different Passes
Through the SPEER System
Second-Level Induced Edits
9,765 Records

Pass	Failed	Passed
First	5,404	4,361
Second	43	9,722
Third	1	9,764

5. **DISCUSSION**

The discussion provides more explanation of the version of SPEER that deals with second-level induced edits and ideas related to imputation. In the following, when we say *correct* a record, we mean to impute new values in a manner so that the record satisfies edits. The intuitive idea of SPEER is that most records will only fail a few edits and are easily corrected by first-level induced edits. The empirical data is quite useful for test purposes because the associated edits contains two balance equations and the two balance equations are sometimes connected by a failing ratio edit. By being *connected*, we mean that one of the two terms in the ratio edit is in one balance equation and the other term is in the second balance equation.

5.1. Second-Level Induced Edit Version of SPEER

Rather than write exceedingly difficult code that would allow SPEER to correct all (or nearly all) of the records on the first pass, we chose to write far simpler code that is easier to maintain and may require several passes to correct a record. The intuitive idea of the code is to correct a record partially on the first pass and finish the corrections on a later pass (preferably the second). Of the 43 records failing the SPEER edits on the second pass, most fail two second-level induced edits and the two failing second-level induced edits are connected by a failing ratio edit. In other words, to assure that we could correct most records on the first pass, we would need to generate implicit edits to at least four or five levels.

5.2. Imputation

When balance equations and other edits must be satisfied, it now appears that determining the minimum number of fields to impute conflicts with maintaining the joint distributions of variables. Kovar and Winkler (1996) provide examples of when ratio imputation can provide slightly better correlations than nearest-neighbor imputation even when the ratio imputation is not imputing the minimum number of fields. Todaro (1997) shows that an earlier version of SPEER that only handles first-level induced edits can perform very poorly when only one item in a balance equation is imputed. He provides examples where one item in a total is missing, the total is large, and the sum of the remaining items is relatively small. By using the balance equation, SPEER can force a large value to be imputed even though prior year data or the data associated with similar records (companies) may suggest that two or more items should be imputed. The difficulty is that if only one item in a balance equation is imputed, then joint relationships between variables are not necessarily maintained. A heuristic solution may be to impute two items when at least one item in a balance equation must be imputed.

This paper reports the results of research and analysis undertaken by Census Bureau staff. It has undergone a Census Bureau review more limited in scope than that given to official Census Bureau publications. This report is released to inform interested parties of research and to encourage discussion.

6. **SUMMARY**

This paper presents theory, algorithms, and results from using the new SPEER edit system that uses the model of Fellegi and Holt (1976). This system is the only one that allows simultaneous editing via ratio inequalities and a limited form of balance equations so that final imputed records satisfy all edits.

REFERENCES

Fellegi, I. P. and Holt, D. (1976), "A Systematic Approach to Automatic Edit and Imputation", *Journal of the American Statistical Association*, **71**, 17-35.

Greenberg, B. G., and Surdi, R. (1984), "A Flexible and Interactive Edit and Imputation System for Ratio Edits", SRD report RR-84/18, U.S. Bureau of the Census, Washington, D.C., USA.

Kovar, J.G., MacMillan, J.H. and Whitridge, P. (1991), "Overview and Strategy for the Generalized Edit and Imputation System", Statistics Canada, Methodology Branch Working Paper BSMD 88-007E (updated in 1991).

Kovar, J.G., and Winkler, W.E., (1996), "Editing Economic Data", *American Statistical Association, Proceedings of the Section on Survey Research Methods*, 81-87.

Thompson, K. J. and Sigman, R. S. (1996), "Statistical Methods for Developing Ratio Edit Tolerances for Economic Censuses", *American Statistical Association, Proceedings of the Section on Survey Research Methods*, 166-171.

Todaro, T. A. (1997), "Adapting the SPEER Edit System to Edit Hog Data in the National Agricultural Statistics Service's Quarterly Agriculture Surveys", *American Statistical Association, Proceedings of the Section on Survey Research Methods*, to appear.

Winkler, W. E., and Draper, L. R. (1997), "New SPEER Edit System", computer system and unpublished documentation, Statistical Research Division, U.S. Bureau of the Census, Washington, D.C., USA.

Winkler, W. E., and Draper, L. R. (1997), "The SPEER Edit System", in *Statistical Data Editing, Volume 2*, U.N. Statistical Commission and Economic Commission for Europe, Geneva, Switzerland, 56-62.